Definitions of Technology Readiness Level (TRL)
For Hardware/Subsystems and Draft for Software

Technology Readiness Level	Description
1. Basic principles observed and reported.	**HW/S**: Lowest Level of Technology Readiness. Scientific research begins to be translated into applied research and development. Examples might include paper studies of a technology's basic properties. **SW**: Lowest level of software readiness. Basic research begins to be translated into applied research and development. Examples might include a concept that can be implemented in software or analytic studies of an algorithm's basic properties.
2. Technology concept and/or application formulated.	**HW/S/SW**: Invention begins. Once basic principles are observed, practical applications can be invented. Applications are speculative and there is no proof or detailed analysis to support the assumptions. Examples are limited to analytic studies.
3. Analytical and experimental critical functions and/or characteristic proof of concept.	**HW/S**: Active research and development is initiated. This includes analytical studies and laboratory studies to physically validate analytical predictions of separate elements of the technology. Examples include components that are not yet integrated or representative **SW**: Active research and development is initiated. This includes analytical studies to produce code that validates analytical predictions of separate software elements. Examples include software components that are not yet integrated or representative but satisfy an operational need. Algorithms run on a surrogate processor in a laboratory environment.
4. Component and/or breadboard validation in laboratory environment.	**HW/S:** Basic technological components are integrated to establish that they will work together. This is relatively "low fidelity" compared to the eventual system. Examples include integration of "ad hoc" hardware in the laboratory. **SW:** Basic software components are integrated to establish that they will work together. They are relatively primitive with regard to efficiency and reliability compared to the eventual system. System software architecture development initiated to include interoperability, reliability, maintainability, extensibility, scalability and security issues. Software integrated with simulated current /legacy elements as appropriate.
5. Component and/or breadboard validation in relevant environment.	**HW/S**: Fidelity of breadboard technology increases significantly. The basic technological components are integrated with reasonably realistic supporting elements so that it can be tested in a simulated environment. Examples include "high fidelity" laboratory integration of components. **SW:** Reliability of software ensemble increases significantly. The basic software components are Integrated with reasonably realistic supporting elements so that it can be tested in a simulated environment. Examples include "high fidelity" laboratory integration of software components. System software architecture established. Algorithms run on a processor(s) with characteristics expected in the operational environment. Software releases are 'Alpha' versions and configuration control initiated. Verification, Validation and Accreditation (VV&A) initiated.
6 System/subsystem model or prototype demonstration in a relevant environment.	**HW/S:** Representative model or prototype system, which is well beyond that of TRL 5, is tested in a relevant environment. Represents a major step up in technology's demonstrated readiness. Examples include testing a prototype in a high fidelity laboratory environment, or in a simulated operational environment. **SW:** Representative model or prototype system, which is well beyond that of TRL 5, is tested in a relevant environment. Represents a major step up in software demonstrated readiness. Examples include testing a prototype in a live/virtual experiment or in simulated operational environment. Algorithm run on processor or operational environment integrated with actual external entities. Software releases are 'Beta' versions and configuration controlled. Software support structure in development. VV&A in process.

Definitions of Technology Readiness Level (TRL)
For Hardware/Subsystems and Draft for Software

7. System prototype demonstration in an operational environment.	**HW/S:** Prototype near, or at, planned operational system. Represents a major step up from TRL 6, requiring demonstration of an actual system prototype in an operational environment, such as an aircraft, vehicle, or space. Examples include testing the prototype in a test bed aircraft. **SW:** Represents a major step up from TRL 6, requiring the demonstration of an actual system prototype in an operational environment, such as in a command post or air/ground vehicle. Algorithms run on processor of the operational environment integrated with actual external entities. Software support structure in place. Software releases are in distinct versions. Frequency and severity of software deficiency reports do not significantly degrade functionality or performance. VV&A completed.
8. Actual system completed and "flight qualified" through test and demonstration.	**HW/S:** Technology has been proven to work in its final form and under expected conditions. In almost all cases, TRL represents the end of true system development. Examples include developmental test and evaluation of the system in its intended weapon system to determine if it meets design specifications. **SW:** Software has been demonstrated to work in its final form and under expected conditions. In most cases, this TRL represents the end of system development. Examples include test and evaluation of the software in its intended system to determine if it meets design specifications. Software releases are production versions and configuration controlled, in a secure environment. Software deficiencies are rapidly resolved through support structure.
9. Actual system "flight proven" though successful mission operations.	**HW/S:** Actual application of the technology in its final form and under mission conditions, such as those encountered in operational test and evaluation. In almost all cases, this is the end of the last "bug fixing" aspects of system development. Examples include using the system under operational mission conditions. **SW:** Actual application of the software in its final form and under mission conditions, such as those encountered in operational test and evaluation. In almost all cases, this is the end of the last "bug fixing" aspects of system development. Examples include using the system under operational mission conditions. Software releases are production versions and configuration controlled. Frequency and severity of software deficiencies are at a minimum.

*Quality attributes include reliability, maintainability, extensibility, scalability and security

ACRONYMs

HW – Hardware
S ----- System/Subsystem
SW – Software